

Octave Notebook

Table of Contents

- Octave Notebook
 - Table of Contents
 - Demonstration Text
 - Preprinted Tabs
 - Syllabus
 - Lab 1
 - General Commands
 - Commands
 - Output
 - Lab 2
 - Variables
 - Commands
 - Output
 - Complex Numbers
 - Commands
 - Output
 - Arrays
 - Commands
 - Output
 - Polynomial Roots
 - Commands
 - Output
 - Plotting
 - Commands
 - Output
 - Examples
 - Square Root
 - Commands
 - Output
 - Population Table
 - Commands
 - Output
 - Average per Game
 - Commands
 - Output
 - Projectile Motion
 - Commands
 - Output
 - Lab 3
 - Question 1
 - Question 2
 - Question 3
 - Question 4
 - Question 5

- Question 6
- Lab 4
 - Exercises Pg 21
 - Commands
 - Output
 - Exercises Pg 25
 - Commands
 - Output
- Lab 5
 - Pg 28 Plot Example
 - Commands
 - Output
 - Pg 32 Explanation
 - Commands
 - Output
 - Pg 29 Exercise 2.3 Lesson 3
 - Commands
 - Output
 - Pg 33 Exercises 2.4 Lesson 4
 - Commands
 - Output
 - Bearing
 - Commands
 - Output
 - Page 37
 - Commands
 - Output
- Lab 6
 - Textbook Problems
 - Commands
 - Output
- Lab 7
 - Script File
 - Commands
 - Output
 - Function File
 - Commands
 - Output
 - Profiling
 - Commands
 - Output
 - Global Variables
 - Commands
 - Output
 - While loops
 - Commands
 - Output
 - Switch, if, keyboard
 - Commands
 - Output
 - Demonstration Script
 - Commands
 - Output
- Lab 8

- Linear Regression
 - Commands
 - Output

Demonstration Text

```
## [Lab 1](#table-of-contents)
### [General Commands](#lab-1table-of-contents)
#### Commands
```matlab
x = 3
y = x + 1
clc # does nothing in script?
clear x
y
6*10/13+18/(5*7)+5*9^2
6*(35^(1/4))+14^0.35
6*(35^(1/4));
x = [-1:0.5...
:0.5];
quit
```
#### Output
```matlab
x = 3
y = 4
y = 4
ans = 410.13
ans = 17.112
x =
 -1.0000 -0.5000 0 0.5000
```

```

Preprinted Tabs

| NOTEBOOK
GRADING SHEET | | | | | ENGR 60 |
|---------------------------|-------|---------------------------------|-------|------------|---------------------|
| | SCORE | | SCORE | NOTES | SCORE |
| Contents | | | | | |
| Cover - Front & Side | | | | | Week 9 Midterm Prep |
| Table of Contents | | | | | Midterm Prep |
| Preprinted Tabs | | | | | Midterm |
| Syllabus | | | | | |
| | | | | p77 | Tab #10 |
| Tab #1 Overview | | Pratap Tab #6 Tutorial 5 | | end of Ch2 | Solvex.m p78-79 |
| | | | | | 4.1 Script files |

| | | NOTEBOOK
GRADING SHEET | | | | ENGR 60 | |
|---------------------------------|--|-----------------------------------|----------------|------------------------------|-----------------------------|------------------------------------|--------|
| T1-1 (a,b) | | 1. On-line help | | prob pg 37 | p80-81 | 4.2 Function Files | XXXXXX |
| Cylinder
20%
increase | | 2. Convert
temperature | | ans pg 38 | Example p82-83 | 4.2.1 Execute a
Function | |
| T1-2 (x+y;
xy; x/y) | | 3.Calculate
Factorials | | | eval w/feval p85 | Evaluating a
Function | |
| 5 sin(u) u(7)
length(w) | | 4. Cross-Product | | | p87-88 | Profiler | |
| T3-1 25th
Element | | 5.Sum Geo series | | | Control-flow p90 | 4.3.4 Loops
Branches | |
| Polynomial
Roots | | 6. Interest
Calculation | | see also
Tab#3
"Money" | for loop (pg 91) | | |
| T3-2
Polynomia
$x^3..290$ | | | | | | Great Bike Race | |
| Plot
$y=\sin(2*x)$ | | Tab #7 | | | | B737 Max 8 | |
| T3-3 Plot
$s=2\sin(3t+2)$ | | Forecasting -
Trend Line | | | | SDC BOOK | |
| T3-4
$y=4\sqrt{6x_1}$ | | | | | | Chapter 1 - Intro
Load | XXXXXX |
| Plot Rocket
Height | | Tab #7 Ch. 3 | | | Matirx input | Load Bearingdata
Save | |
| Script
SampleSQRT | | Ch. 3 How-to | | p41-42 | Matrix manipulation | Basic Statistical
Function | |
| Script
PopTable | | 3.1.1 Input &
Continuation | p45 Fig
3.1 | EXAMPLE | Running the Script
Vt | | |
| Script
Powers of 3 | | 3.1.2
Indexing/Subscript | | p.44 | Continuation . . . | Converting Values | |
| Script A/g
per Game | | 3.1.2 Dimension | | p46 | Input: Vector | Chapter 2 -
Programming | XXXXXX |
| Narrative | | 3.1.3 Matrix
manipulation | p47 | Matrix
Reshaping | Arithmetic/Relational | | |
| | | Transpose -
Initiialization | | p.48 | size(A) | Logical-Variable
Naming | |
| Tab #2
Matrix Math | | Appending -
Deleting | | p.49 | | Storing Numeric
Values | |
| 4 hr Video
(Hr 1 Vectors) | | Utility - Special | | p.50 | zeroes(1,10);
ones(10,1) | Constants/Numerical
Func | |

| | | NOTEBOOK
GRADING SHEET | | | | ENGR 60 | |
|---------------------------------|------------------------------|-----------------------------------|-------------|--------------|--|--------------------------------|--------|
| Matlab Coursework Notes | | Utility Matrix example | | p.51 Fig 3.2 | EXAMPLE eye(3)
diag(B) | Strings/Char | |
| | | 3.1.4 Creating Vecors | | p52 | linspace(0,20,5)
logsp | Import Genetic Data | |
| Pratap Tab #2 Tutorial 1 | | 3.2.1 Matrix & Array Op | XXXXXX | p53 | .* .\ ./ .^ .' | Chapter 3 - Programming | XXXXXX |
| 1. compare;
sqrt; pi | | 3.2.2 Relational Ops | XXXXXX | p54 | <,<=,>,>=,==,~= | Array Types | |
| 2. Exponents & Log (a,b,c) | Matrix transpose,
. * . ^ | | p55 Fig 3.3 | EXAMPLE | Strings as Matrices | | |
| 3. Trig (a, b, c) | | | | | | : ; randi() magic() | |
| 4. Complex No. (a,b,c) | | Tab #8 | | | | Cell Arrays - Structures | |
| | | 3.2.3 Logical Ops | | p56 | & (and) | (or) ~ (not) | Tables |
| Pratap Tab #3 Tutorial 2 | | 3.2.4 Trig Functions | | p57 | sin cos tan cot asin | Chapter 4 - Programming | XXXXXX |
| 1. Equation of Line | | exponent & complex func | p58 | | While Loops | | |
| 2. Vectors (a,b,c) | | round off functions | | p59 | | For Loops | |
| 3. Points on Circle | | 3.2.5 Matrix functions | | p59 | | If - Else Conditionals | |
| 4. Geometric Series | | 3.2.6 Character strings | | p59,61 | nam=
'Doe'; 'Ravi'"
'Doe';' Ravi'' | If - Else - If Swith- Case | |
| 5. Matrices & Vectors | | Matrix function vs Array | | p60 Fig 3.4 | EXAMPLE | Editor TerminalVelocity | |
| | | Manipulating char strings | | p61 | | Image File - Pretty | |
| Tab #3 | | Eval function | | p.62-63 | strcmp, num2str | Exiting Stopping a Loop | |
| Time Value Money CANVAS | 3.3 Functions and Help | | p64 | | Chapter 5 - Matrices | XXXXXX | |
| | | Help Directory | XXXXXX | p65 Fig 3.5 | Size, diag, eye, magic | | |
| Pratap Tab #4 Tutorial 3 | | determinant help, eig | XXXXXX | p66-69 | | Scalar math - .* ./ | |

| | | NOTEBOOK
GRADING SHEET | | | ENGR 60 | |
|---------------------------------|--|-----------------------------------|--------|------------------------|-------------------------------------|--------|
| 1. Simple Sin Plot | | | XXXXXX | | Transpositions | |
| 2. Line Styles | | | XXXXXX | | Exponents, Logical ops | |
| 3. Decay Sin Plot | | | XXXXXX | | Chap 6 - Function, Script | XXXXXX |
| 4 Space Curve | | | | | Script - Interactive | |
| 5. On-line Help | | 3.4 Saving Loading Data | XXXXXX | | Functions Built-in | |
| 6. Log scale Plots | | 3.4.1 Loading mat files | XXXXXX | | Fuction - Format | |
| 7. Overlay Plots | | | | | Function Algorithm Code | |
| 8. 9. Own Plots | | 3.4.2 Recording w/diary | | p70 | Scope - Recursion | |
| | | 3.5 Plotting | | p71-73 | Chap 7 - Debugging | XXXXXX |
| Pratap Tab #5 Tutorial 4 | | | | | Debugger | |
| 1. Center of Circle | | Matrix Exercises | | p.74 | Break Points | |
| 2. Change radius of Circle | | 1. Entering Matrices | | | Error Handle - Comments | |
| 3. Variables in workspace | | 2.Linear Algebra Rules | XXXXXX | | Chap 8 - Import/Export | XXXXXX |
| 4. Contents of file "type" | | 3. Create matrices | | | ls mkdir * path | |
| 5. H1 line | | 4.Matrices -Sub-matrices | | | importing - exporting | |
| 6. Just for Fun script file | | 5.Manipulate a matrix | | | Chap 9 - Plotting & Data | XXXXXX |
| Bearing.mat CANVAS | | 6.See the structures | | | Plotting | |
| Your Own Project | | 7. Create a symmetric matrix | | Chap 10 - Tools | XXXXXX | |
| | | 8. Do some cool ops | | | Tools | |
| | | | | | Chap 11 - Symbolic Math | |

| | NOTEBOOK
GRADING SHEET | | | | ENGR 60 | |
|------------------------|---------------------------|--|--|--|-----------------------|--|
| TOTAL
COLUMN
ONE | TOTAL COLUMN
TWO | | | | TOTAL COLUMN
THREE | |

Syllabus

Syllabus



ENGR 060 Program and Problem-Solving in MATLAB

Section 32846, Fall 2020

How to Contact Me

| Type | Information |
|----------------------------|--|
| Instructor | Instructor: Thomas J. Sanford, P.E. |
| Faculty Web Page | https://www.westvalley.edu/faculty/sanford_t.html |
| Phone | 408-741-4608 |
| Email | Use Canvas email |
| Office Location | Zoomlink: https://cccconfer.zoom.us/meeting/register/tJrcqeagpzguGtH1oD6qrAWDirY47XISuA#
password 367535 |
| Office Hours | 7:00 – 7:45 am |
| Class Days, Time, Location | MW 7:45 – 10:45 |
| Course Format | Online |

Welcome to ENGR 60 MATLAB?

a course that should help you accomplish your goals in pursuing an engineering, a physics, a science or math education.

Course Information

Catalog Description: This course utilizes the MATLAB environment to provide students with a working knowledge of computer-based problem-solving methods relevant to science and engineering. It introduces the fundamentals of procedural programming, numerical analysis, and data structures. Examples and assignments in the course are drawn from practical applications in engineering, physics, and mathematics.

Student Learning Outcomes: Solve engineering problems via MATLAB using a symbolic language to deliver a correct solution.

Required Texts and Materials

- An Engineer's Introduction to Programming with MATLAB 2019 by Shawna Lockhart, Eric Tilleson; Published June 17, 2019, ISBN 978-1-63057-292-1
- MATLAB software OR Download free shareware called OCTAVE GNU

Assignments

You'll start each week with reading the chapter. You'll do lab(s), notes, quiz(s); You start on Monday and finish by Saturday at 11 pm.

Notebook



Weekly homework requires a recording all your transactions to a notebook. See the Notebook checksheet and schedule for the due date. Student samples are provided as to how to organize it.

Grading

| <u>Exercises</u> | <u>Points</u> |
|--|---------------|
| Labs, Quizzes, Notebook, Homework..... | 50 |
| Exams..... | <u>50</u> |
| Total Possible..... | 100 |

Grade Distribution

| Number of | | | |
|---------------|--------------|-------------------|---------------------|
| <u>Points</u> | <u>Grade</u> | <u>Definition</u> | <u>Grade Points</u> |
| 90-100 | A | Superior | 4.0 |
| 80-89 | B | Above Average | 3.0 |
| 70-79 | C | Average | 2.0 |
| 60-69 | D | Passing | 1.0 |
| 59 OR LESS | | Failure | 0 |

West Valley College Policies

It's important for you to be aware of policies in the [West Valley College Catalog](#). Below are a few key policies from the *Rights and Responsibilities* section you should know about.

- **Attendance:** Know that instructors may drop any student who does not attend the first class, who has missed any one class during the first three weeks of instruction, or who has missed more than ten percent of the total class hours for the semester.
- **Registration and Drops:** Understand that it is your responsibility to make sure you are registered and/or dropped from any class by the deadlines listed in the schedule of classes.
- **Withdrawal from a class:** Make sure you know the last "day to drop with a W." (Put that date in your calendar!) Even if you ask an instructor to drop you, it is still your responsibility, so double-check your enrollment status in the Portal. It's important to know that instructors are *required to issue a grade* for any student still enrolled after the last day to drop.
- **Academic Dishonesty:** Understand what this means--it includes in-class cheating, out-of-class cheating, plagiarism, helping another student in cheating or plagiarism, or knowingly giving false information to College staff, faculty, administrators or other officials. (See the *Rights and Responsibilities, Academic Dishonesty* section for more information and further definitions.)

Important Dates

- Last day to ADD this class: Thursday, 10 Sep 2020
- Last day to DROP this class without a "W" and with a refund: Sunday 13 Sep 2020



- Last day to DROP this class with a "W:" 20 Nov 2020
- FINAL EXAM for this class: 14 Dec 2020

West Valley College Nondiscrimination Statement

The District, and each individual who represents the District, shall provide access to its services, classes and programs without regard to national origin, religion, age, sex or gender, race, color, medical condition, ancestry, sexual orientation, marital status, physical or mental disability, or because he/she is perceived to have one or more of the foregoing characteristics, or based on association with a person or group with one or more of these actual or perceived characteristics.

Accommodations for a Disability

West Valley College strives to make all learning experiences as accessible as possible. If you anticipate or experience academic barriers based on your disability (including permanent disabilities or chronic or temporary medical conditions), please let me know immediately so that we can privately discuss options. You are also welcome to register with the Disability and Educational Support Program to establish reasonable accommodations. After registration, DESP will notify of your approved accommodations.

Disability and Educational Support (DESP) Contact Information

The DESP office is located in the LS building. Their phone number is (408) 741-2010 (voice) or (408) 741-2658 (TTY). Information about their services can be found at westvalley.edu/desp.

Student Help and Support

West Valley College has many Student Services that are here to support your success and help you achieve your goals. Check them out at westvalley.edu/services.

Emergency Information

Be prepared! Review the college's Emergency Preparedness information at westvalley.edu/emergency. It is recommended putting WVC emergency numbers into your phone.

Special Notes:

1. Regular Attention/Connection to Class is key to educational success. Random rolls taken.
2. Late work/non-continuous contact loss results: 10% cut then 50% after answers revealed.
3. Study all the required chapters for tests, independent of the lecture pace.
4. No video/audio (tape) recording without prior written instructor permission.
5. Syllabus subject to change without notice due to changing conditions of course. Check often.
6. Internet to be used for the purpose of education in classroom topics only.
7. Problems: first talk to the instructor, detail problem in writing, then schedule an appointment.

Lab 1

General Commands

Commands

```
x = 3
y = x + 1
clc # does nothing in script?
clear x
y
6*10/13+18/(5*7)+5*9^2
6*(35^(1/4))+14^0.35
6*(35^(1/4));
x = [-1:0.5...
:0.51]
quit
```

Output

```
x = 3
y = 4
y = 4
ans = 410.13
ans = 17.112
x =
-1.0000 -0.5000      0    0.5000
```

Lab 2

Variables

Commands

```
r = 8;
h = 15;
V = pi * r ^ 2 * h;
V = V + 0.2 * V;
r = sqrt(V/(pi*h))
```

Output

```
r = 8.7636
```

Complex Numbers

Commands

```
x = -5 + 9i;
y = 6 - 2i;
whos
x + y
x*y
x/y
```

Output

```
Variables visible from the current scope:
```

```
variables in scope: top scope
```

| Attr | Name | Size | Bytes | Class |
|-------|-------|-------|-------|--------|
| ===== | ===== | ===== | ===== | ===== |
| c | x | 1x1 | 16 | double |
| c | y | 1x1 | 16 | double |

```
Total is 2 elements using 32 bytes
```

```
ans = 1 + 7i
ans = -12 + 64i
ans = -1.2000 + 1.1000i
```

Arrays

Commands

```
u = [0: 0.1 :10];
w = 5 * sin(u);
u(7)
w(7)
m = length(w)
v = cos(0): 0.02 :log10(100);
length(v)
v(25)
```

Output

```
ans = 0.6000
ans = 2.8232
m = 101
ans = 51
ans = 1.4800
```

Polynomial Roots

Commands

```
a = [1,-7,40,-34];
roots(a)
```

Output

```
ans =
3.0000 + 5.0000i
3.0000 - 5.0000i
1.0000 + 0i
```

Plotting

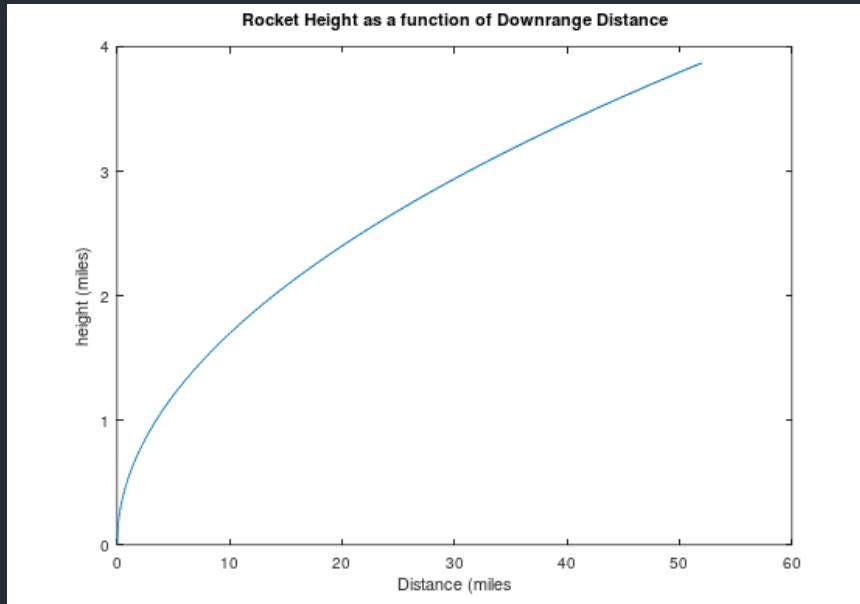
Commands

```

x = [0:0.1:52];
y = 0.4*sqrt(1.8*x);
plot(x,y), xlabel('Distance (miles'), ylabel('height (miles)'),...
title('Rocket Height as a function of Downrange Distance')
pause

```

Output



Examples

Square Root

Commands

```

x = [1:10];
y = sqrt(x)

```

Output

```

y =

Columns 1 through 8:

1.0000    1.4142    1.7321    2.0000    2.2361    2.4495    2.6458    2.8284

Columns 9 and 10:

3.0000    3.1623

```

Population Table

Commands

```

yr = [1984 1986 1988 1990 1992 1994 1996];
pop = [127 130 136 145 158 178 211];
tableYP(:,1) = yr';
tableYP(:,2) = pop';
disp('')
disp('    YEAR      POPULATION')
disp('                      (MILLIONS)')
disp('')
disp(tableYP)
disp('')

```

Output

| YEAR | POPULATION
(MILLIONS) |
|------|--------------------------|
| 1984 | 127 |
| 1986 | 130 |
| 1988 | 136 |
| 1990 | 145 |
| 1992 | 158 |
| 1994 | 178 |
| 1996 | 211 |

Average per Game

Commands

```
number1 = input('Enter the points scored in the first game: ');
if number1 < 0
    disp('Warning! Input Invalid. Using absolute value.')
    number1 *= -1;
end
number2 = input('Enter the points scored in the second game: ');
if number2 < 0
    disp('Warning! Input Invalid. Using absolute value.')
    number2 *= -1;
end
number3 = input('Enter the points scored in the third game: ');
if number3 < 0
    disp('Warning! Input Invalid. Using absolute value.')
    number3 *= -1;
end

disp('The average of points scored is ')
disp((number1 + number2 + number3)/3)
```

Output

```
Enter the points scored in the first game: -1
Warning! Input Invalid. Using absolute value.
Enter the points scored in the second game: 2
Enter the points scored in the third game: 0
The average of points scored is
1
```

Projectile Motion

Commands

```
speed = input('Speed: ');
gravity = -9.81;
startHeight = input('Starting Height: ');
angle = input('Starting Angle in Radians: ');
disp('');
angle
horizontalVelocity = speed * cos(angle)
verticalVelocity = speed * sin(angle)
timeInFlight = verticalVelocity/(abs(gravity/2))
horizontalVelocity = timeInFlight * horizontalVelocity
disp('');
```

Output

```

Speed: 50
Starting Height: 0
Starting Angle in Radians: pi/4

angle = 0.7854
horizontalVelocity = 35.355
verticalVelocity = 35.355
timeInFlight = 7.2080
horizontalVelocity = 254.84

```

Lab 3

Question 1

| Enter your House Information here. | | | |
|------------------------------------|-----------------|------|--------|
| Style | Condo | BD | 2 |
| County | Santa Clara, CA | TB | 2 |
| Sq. Feet | 950 | Acrq | 0.0219 |

Question 2

| 15-Year Mortgage Loan Details | |
|-------------------------------|------------|
| Select a Bank Option: | Bank 1 |
| Closing Costs Fees: | \$1,400.00 |
| Closing Costs Points: | 1.0 |
| Loan Interest Rate: | 3.890% |

| 30-Year Mortage Loan Details | |
|------------------------------|------------|
| Select a Bank Option: | Bank 1 |
| Closing Costs Fees: | \$1,700.00 |
| Closing Costs Points: | 1.3 |
| Loan Interest Rate: | 4.59% |

Question 3

| House Listing Price | \$849,000.00 |
|---------------------|--------------|
| Down Payment | \$170,000.00 |
| Closing Costs | \$8,190.00 |
| Total Loan Amount | \$687,190.00 |
| APR | 3.89% |
| Loan Term (months) | 180 |
| Monthly Payment | \$5,045.26 |
| Total Payments | \$908,147.57 |

House Listing Price**\$849,000.00**

Total Interest

\$220,957.57

Question 4

| House Listing Price | \$849,000.00 |
|----------------------------|---------------------|
| Down Payment | \$170,000.00 |
| Closing Costs | \$10,187.50 |
| Total Loan Amount | \$689,187.50 |
| APR | 4.59% |
| Loan Term (months) | 360 |
| Monthly Payment | \$3,528.96 |
| Total Payments | \$1,270,426.79 |
| Total Interest | \$581,239.29 |

Question 5

| 15-year vs. 30-year Mortgage Comparison | |
|--|------------------|
| Which loan has the higher monthly payment? | 15-year Mortgage |
| How much higher per month is the payment? | \$1,516.30 |
| Which loan has more total interest paid? | 30-year Mortgage |
| How much more total interest is paid? | \$360,281.71 |

| Equity Comparison | |
|--|--------------|
| Suppose you sell your home at year 10 for \$1018800. After you repay the remaining balance of your home, any money you have left over is called equity. The following questions help you compare the equity in your home after 10 years. | |
| 15-year loan ending balance at year 10? | \$274,692.95 |
| 30-year loan ending balance at year 10? | \$553,547.39 |
| 15-year loan equity at year 10? | \$744,107.05 |
| 30-year loan equity at year 10? | \$465,252.61 |

Question 6

While the 15-year mortgage seems objectively better as I would have to pay less in the long term, if I were to purchase the house right now, I would be unable to pay the higher monthly payments. Instead of purchasing a house immediately, I could try to save money now and later take out a 15-year loan.

Lab 4**Exercises Pg 21****Commands**

```

2^(5/(2^5-1))
(1-1/2^5)^1
3*(sqrt(5)-1)/(sqrt(5)+1)^2-1
r = pi^(1/3)-1
pi*r^2

disp('')

exp(3)
log(exp(3))
log10(exp(3))
log10(10^5)
exp(pi*sqrt(163))
fsolve(@(x) [3^x-17],0)
log(17)/log(3)

disp('')

sin(pi/6)
cos(pi)
tan(pi/2)
sin(pi/6)^2 + cos(pi/6)^2
x = 32*pi;
y = cosh(x)^2-sinh(x)^2

disp('')

(1+3i)/(1-3i)
e^(i*pi/4)
cos(pi/4)+i*sin(pi/4)
exp(pi/2*i)
exp(pi/2i)
disp('pi/2i => pi/(2*i) != pi/2*i')

```

$$\frac{1+3i}{1-3i} = \frac{(1+3i)(1+3i)}{(1-3i)(1+3i)} = \frac{1+6i-9}{1-(-9)} = \frac{-8+6i}{10} = -0.8+.6i$$

$$1 - 3i + 3i = (1 - 3i)(1 + 3i)(1 + 3i) = 1 - (-9)1 + 6i - 9 = 10 - 8 + 6i = -0.8 + .6i$$

Output

```

ans = 1.0323
ans = 0.9688
ans = -0.6459
r = 0.4646
ans = 0.6781

ans = 20.086
ans = 3
ans = 1.3029
ans = 5
ans = 2.6254e+17
ans = 2.5789
ans = 2.5789

ans = 0.5000
ans = -1
ans = 1.6331e+16
ans = 1
y = 0

ans = -0.8000 + 0.6000i
ans = 0.7071 + 0.7071i
ans = 0.7071 + 0.7071i
ans = 6.1232e-17 + 1.0000e+00i
ans = 6.1232e-17 - 1.0000e+00i
pi/2i => pi/(2*i) != pi/2*i

```

Exercises Pg 25

Commands

```
y = @(x) [0.5*x-2];
y(0), y(1.5), y(3), y(4), y(5), y(7), y(9), y(10)

disp(' ');

t = [1:10];
x = t.*sin(t)
y = (t-1)./(t+1)
z = sin(t.^2)./t.^2

disp(' ');

theta = [0; pi/4; pi/2; 3*pi/4; pi; 5*pi/4];
r = 2;
x = r*cos(theta)
y = r*sin(theta)
sqrt(x.^2 + y.^2)

disp(' ');

n = [0:10];
r = 0.5;
sum(r.^n)
1/(1-0.5)
n = [0:50];
sum(r.^n)
n = [0:100];
sum(r.^n)

disp(' ');

# no page 45 provided
v = 0:0.2:12;
M = [sin(v); cos(v)];
size(v), size(M)
M(1, 1:10)
M(2, 1:10)
```

Output


```

ans = 2.0000
ans = 2

ans =
1   61

ans =
2   61

ans =
Columns 1 through 8:

0    0.1987    0.3894    0.5646    0.7174    0.8415    0.9320    0.9854

Columns 9 and 10:

0.9996    0.9738

ans =

Columns 1 through 7:

1.000000    0.980067    0.921061    0.825336    0.696707    0.540302    0.362358

Columns 8 through 10:

0.169967   -0.029200   -0.227202

```

Lab 5

Pg 28 Plot Example

Commands

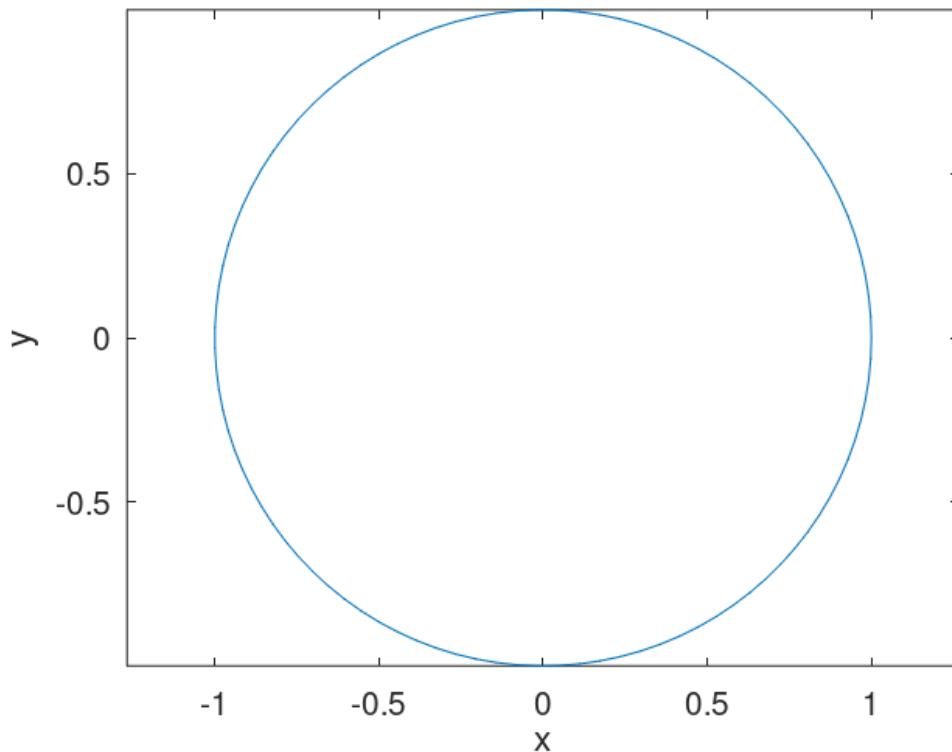
```

theta = linspace(0,2*pi,100);
x = cos(theta);
y = sin(theta);
plot(x,y)
axis('equal');
xlabel('x')
ylabel('y')
title('Circle of unit radius')
print # Did not make any action
pause; # So the program does not exit immediately.

```

Output

Circle of unit radius



Pg 32 Explanation

Commands

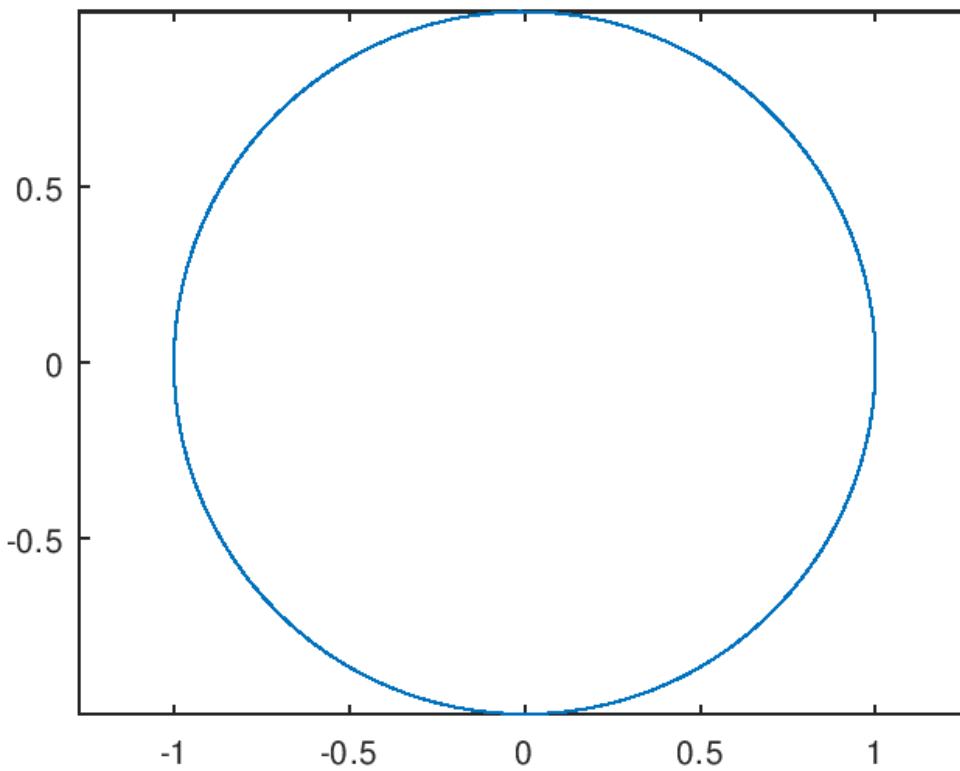
```
% CIRCLE - A script file to draw a unit circle
% File written by Rudra Pratap. Last modified 6/28/98
%
theta = linspace(0,2*pi,100); % create vector theta
x = cos(theta); % generate x-coordinates
y = sin(theta); % generate y-coordinates
plot(x,y);
axis('equal');
title('Circle of unit radius')
```

Output

```
octave:1> help circle
'circle' is a script from the file /Users/francischua/gitprojects/octave/circle.m

CIRCLE - A script file to draw a unit circle
File written by Rudra Pratap. Last modified 6/28/98
```

Circle of unit radius



Pg 29 Exercise 2.3 Lesson 3

Commands

```
x = linspace(0,2*pi,100);
y = sin(x);
plot(x,y); axis('equal');
title('Plot created by Francis'); xlabel('x'); ylabel('y');
pause
plot(x,y,'o'); axis('equal');
pause
plot(x,y,x,y,'o'); axis('equal');
pause;

theta = linspace(0,4*pi,100);
y = exp(-0.4*theta).*sin(theta);
axis([-1, 4*pi, -0.5, 10]);
plot(theta,y);
pause;

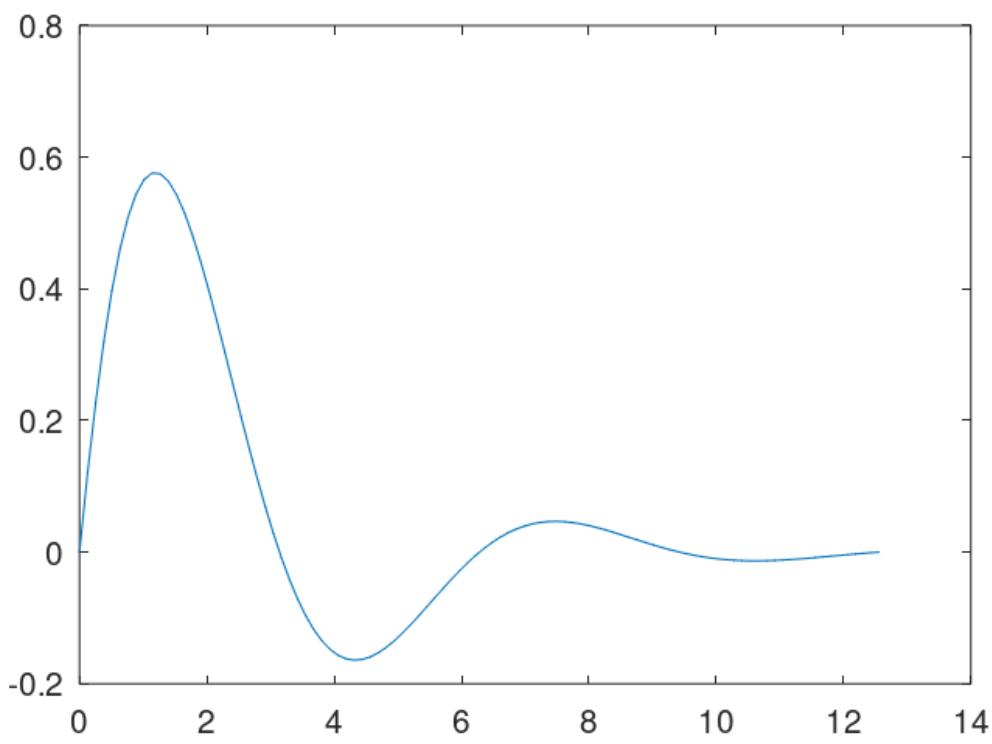
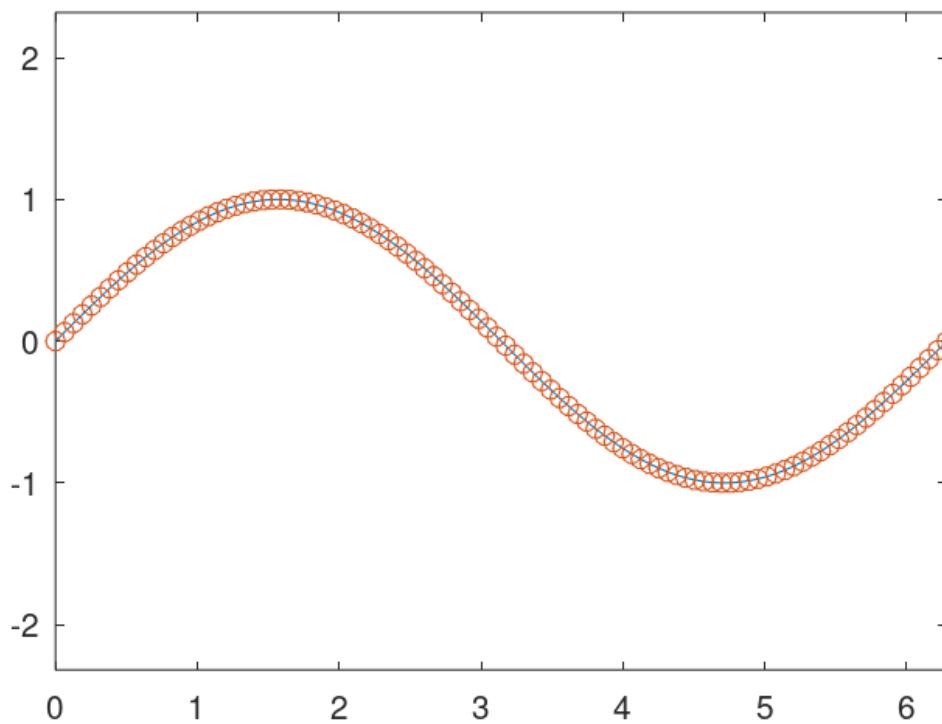
t = linspace(0,20,100);
x = sin(t); y = cos(t); z = t;
plot3(x,y,z);
pause;

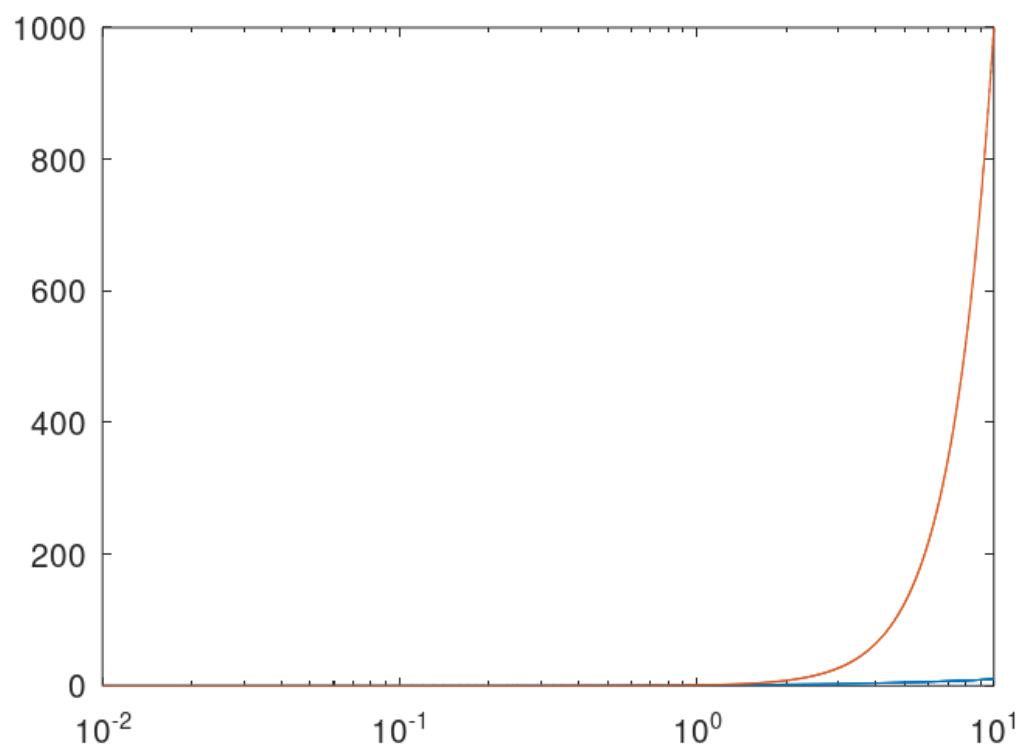
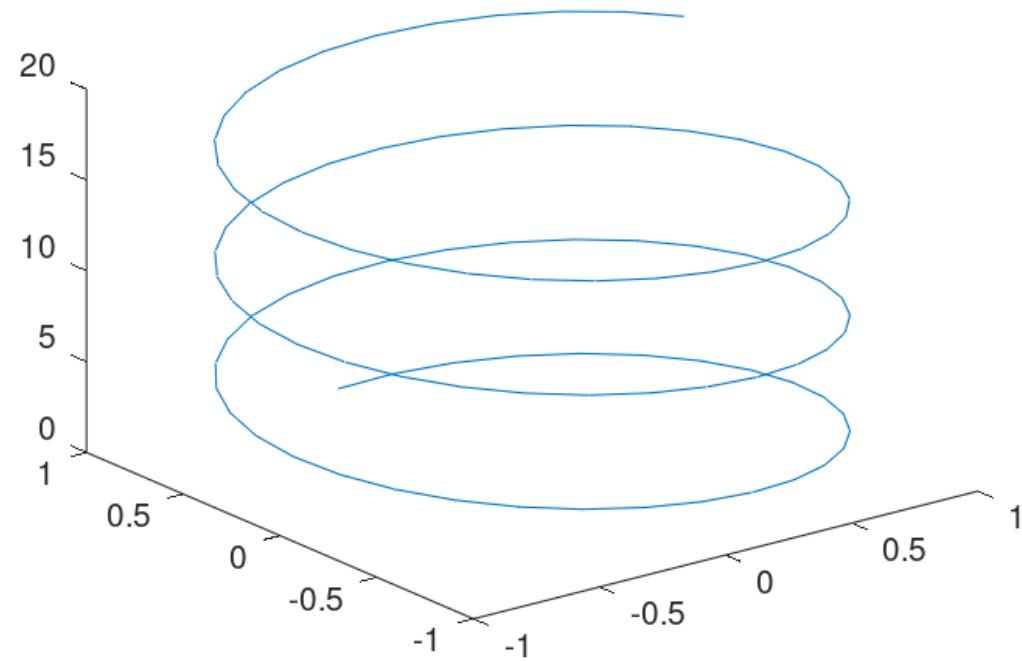
help plot

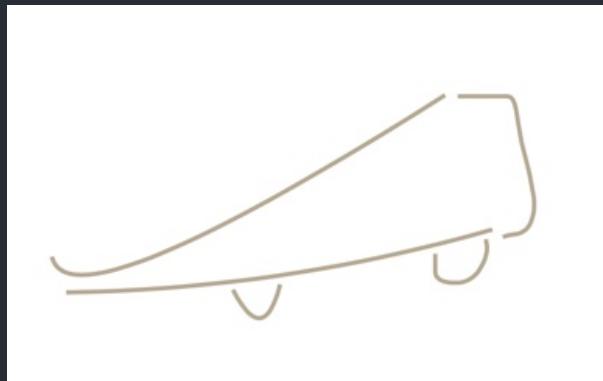
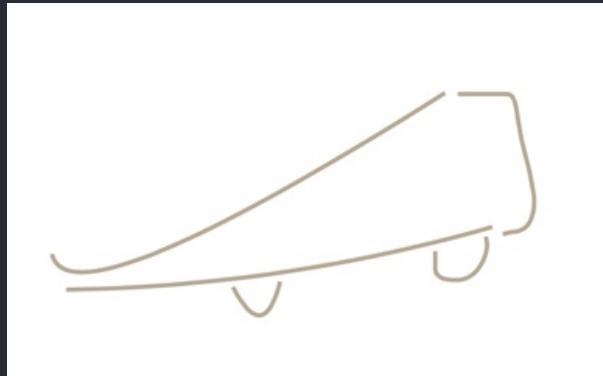
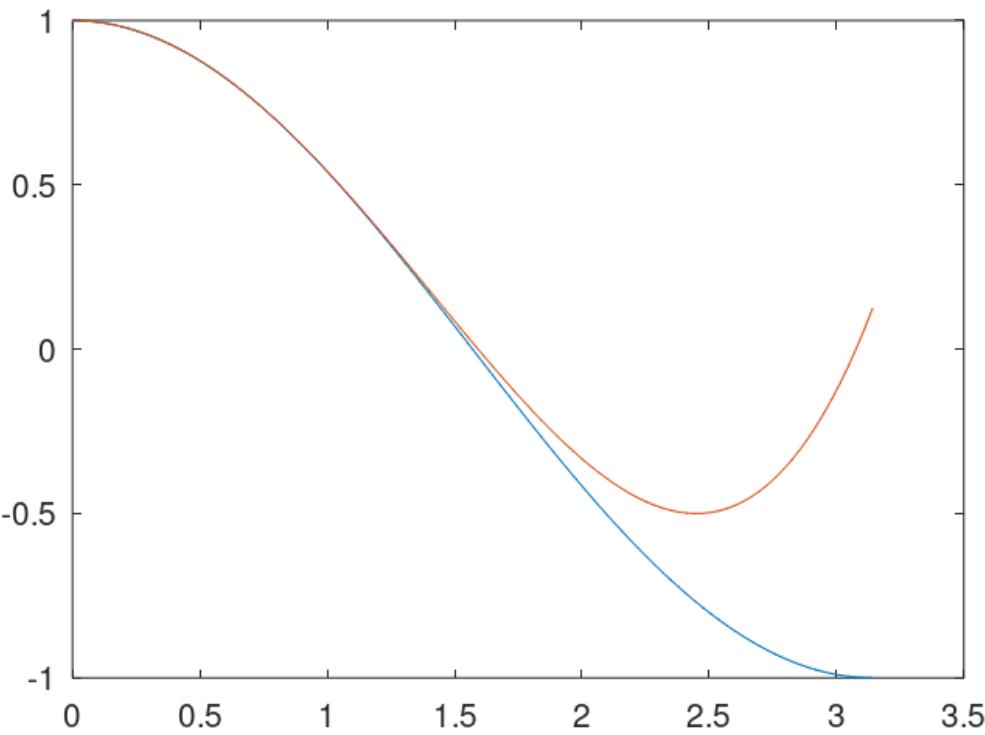
x = linspace(0,10,1000);
y = x.^3;
semilogx(x,x,x,y);
pause;

x = linspace(0,pi,100);
plot(x,cos(x),x,1-x.^2/2+x.^4/24)
pause
```

Output







Pg 33 Exercises 2.4 Lesson 4

Commands

```
circle  
who  
whos  
[theta' x' y']  
type circle.m  
lookfor unit  
  
strftime ("%r %A %e %B %Y", localtime (time ()))  
disp('___ is the cursed teacher.')
```

Output


```
luupdate          Given an LU factorization of a real or complex matrix A = L
                  *U, L lower unit trapezoidal and U upper trapezoidal, return
                  the LU factorization of A + X*Y.

info              Display contact information for the GNU Octave community.

cylinder         Plot a 3-D unit cylinder.

sphere            Plot a 3-D unit sphere.

addtodate        Add Q amount of time (with units F) to the serial datenum,
                  D.

circle            CIRCLE - A script file to draw a unit circle File written
                  by Rudra Pratap.

ans = 05:39:07 PM Thursday 11 March 2021
____ is the cursed teacher.
```

Bearing

Commands

```
load('assets/bearingdata.mat')
min(data)
bounds(data) # only does lowest
max(data)
mean(data)
median(data)
mode(data)
std(data)
var(data)
corrcorcoef(data)
cov(data)
```

Output

```
ans = 2.3601
ans = 2.3601
ans = 2.3650
ans = 2.3623
ans = 2.3623
ans = 2.3601
ans = 1.4878e-03
ans = 2.2135e-06
ans = 1
ans = 2.2135e-06
```

Page 37

Commands

```

help disp
function temp = convert(Ti, Tf)
    C = [Ti: Tf]';
    F = 9/5*C + 32
    temp = [C F];
end
convert(-1, 10)

function output = fact(n)
    output = 1;
    while n > 1
        output *= n;
        n--;
    endwhile
end

fact(4)
fact(9)

function out = crossprod (u, v)
    out = [u(2)*v(3) - u(3)*v(2), u(3)*v(1) - u(1)*v(3), u(1)*v(2) - u(2)*v(1)];
end

crossprod([1,0,0], [0,1,0])
crossprod([0,1,0], [0,0,1])

function out = summer(r, n)
    out = (r^(n+1)-1)/(r-1);
end

summer(0.5, 5)
summer(1/3, 2)

function out = interest(X, n, r, k)
    out = X * ((1+r/k)^(k*n)-1);
end

format bank
interest(1000,5,.06, 4)
interest(1000,5,.06, 365)

```

Output


```
ans = 346.86  
ans = 349.83
```

Lab 6

Textbook Problems

Commands


```
A = ones(20);
A(6:15,6:15) = zeros(10);
spy(A) # bunch of stars with empty square in middle
A(16:20,1:5) = zeros(5);
A(1:5,16:20) = zeros(5);
spy(A)

A = diag(1:6) + diag(7:11,1) + diag(12:15,2)
A = A + triu(A,1)'

A = rand(10);
A = fix(A*100)
idx = A < 10;
A(idx) = 0;
idx = A > 90;
A(idx) = inf
idx = (A <= 50 & A >= 30);
b = A(idx)'
```

Output


```

24   78   26    5   83   85    9   16   86   52
75   13   56   57   84    9   79   48   34   91
12   38   62   50   11   22   62   39   42    2

```

A =

```

 0     0    43    54    38    70    38    56    20    10
 43     0   Inf    53    42    10   Inf    67    39    14
 88    75    22     0   31    41   Inf    21    83    65
 18    56     0   86    54     0   Inf     0    27    15
 20    23     0   43    81    55   Inf    65    83    18
 Inf   Inf    76   Inf    44   Inf     0    26    65     0
 11    78    43   Inf    23   Inf    73    49    27    35
 24    78    26     0   83    85     0    16    86    52
 75    13    56    57   84     0    79    48    34   Inf
 12    38    62   50   11    22   62    39    42     0

```

b =

```

43   38   43   43   43   50   38   42   31   44   41   38   49   48   39   39   34   42   35

```

Lab 7

Script File

Commands

```
%----- This is the script file 'solvex.m' -----
% It solves equation (4.1) for x and also calculates det(A).
```

```
A = [2 2*r r; 3 6 2*r-1; 2 r-1 3*r];
b = [2; 3; 5];
det_A = det(A);
x = A\b;
```

```
%----- CLI Commands -----
r = 1;
solvex
```

Output

```
det_A = 10.000
x =
-0.2000
 0.3000
 1.8000
```

Function File

Commands

```
%----- This is the script file 'solvex.m' -----
function [det_A, x] = solvexf(determinant_func, r);
% SOLVEXF solves a 3x3 watrix equation with parameter function
% To call this function, type:
% [det_A,x] = solvexf(determinant_func, r);
% r is the input and det_A and x are output
%
A = [2 2*r r; 3 6 2*r-1; 2 r-1 3*r];
b = [2; 3; 5];
det_A = feval(determinant_func,A)
x = A\b

%----- CLI Commands -----
[detA, x] = solvexf(@det, 1)
```

Output

```
det_A = 10.000
x =
-0.2000
0.3000
1.8000
```

Profiling

Commands

```
profile on
[detA, x] = solvexf(@det, 1)
profile off
profshow % shows the contents of the current profiling session
```

Output

```
detA = 10.000
x =
-0.2000
0.3000
1.8000
```

| # | Function Attr | Time (s) | Time (%) | Calls |
|----|---------------------|----------|----------|-------|
| 9 | profile | 0.000 | 23.76 | 1 |
| 1 | solvexf | 0.000 | 21.38 | 1 |
| 7 | display | 0.000 | 16.70 | 2 |
| 8 | disp | 0.000 | 12.19 | 2 |
| 5 | det | 0.000 | 11.84 | 1 |
| 4 | feval | 0.000 | 6.01 | 1 |
| 6 | binary \ | 0.000 | 4.06 | 1 |
| 12 | false | 0.000 | 1.50 | 1 |
| 2 | binary * | 0.000 | 0.80 | 3 |
| 10 | nargin | 0.000 | 0.71 | 1 |
| 11 | binary != | 0.000 | 0.71 | 1 |
| 3 | binary - | 0.000 | 0.35 | 2 |
| 13 | __profiler_enable__ | 0.000 | 0.00 | 1 |

Global Variables

Commands

```
%----- Script File -----
global k_value c_value;

function xdot = ode1(t,x);
% ODE1: function to compute the derivative xdot
global k_value c_value
xdot = k_value * x + c_value * sin(t);
end

k_value = 5; c_value = 2;
ts = [0 20];
x0 = 1.0;
[t, x] = ode23('ode1', ts, x0)
[t(1:10), x(1:10)]
```

Output

```
ans =

    0    1.0000
  0.0178    1.0933
  0.0445    1.2510
  0.0845    1.5337
  0.1330    1.9662
  0.1841    2.5561
  0.2359    3.3370
  0.2881    4.3618
  0.3405    5.7031
  0.3930    7.4565
```

While loops

Commands

```
v = 1; i = 1; num = 0;
while num < 10000
    num = 2^i;
    v = [v; num];
    i = i + 1;
end
v
```

Output

```
v =

 1
 2
 4
 8
16
32
64
128
256
512
1024
2048
4096
8192
16384
```

Switch, if, keyboard

Commands

```
%----- This is the script file 'work.m' -----
A = ones(10)
for i = 1:10
    disp(i)
    if i == 5
        keyboard
    elseif i==4
        for j=1:10
            switch mod(A(3,j),2)
                case 0
                    A(4,j) = 1;
                case 1
                    A (4,j) = -2;
            end
        end
    end
    A(:,i) = i*A(:,i);
end
%----- CLI Commands -----
work
keyboard> A
keyboard> return
A
```

Output

Demonstration Script

Commands

```

function [det_A, x] = solvexf(determinant_func);
% SOLVEXF solves a 3x3 watrix equation with parameter function

r = input('r: ');
A = [5 2*r r; 3 6 2*r-1; 2 r-1 3*r]
b = [2;3;5];
det_A = feval(determinant_func, det(A)) % uses a user defined function to compute determinant
x = A\b;
gain_control = input('Would you like to gain control? (y/n) ','s');
switch gain_control
    case 'y'
        disp('Type "return" and hit RETURN to continue.');
        keyboard;
    case 'n'
        choice = menu('What would you like to do now?','Sleep','print ''x'''');
        switch choice
            case 1
                disp('Goodnight')
                exit
            case 2
                x
                disp('.')
                pause(1)
                disp('.')
                pause(1)
                disp('.')
                pause(1)
                disp('GO SLEEP')
                exit
            end
        otherwise
            error('Invalid Choice');
    end
    disp('Continuing')
FallSem(1).course = 'cs101';
FallSem(1).prof = 'turing';
FallSem(1).score = [80 75 95];
FallSem
FallSem(2).course = 'phy200';
FallSem(2).prof = 'Fiegenbaum';
FallSem(2).score = [72 75 78];
FallSem
FallSem(2).course
container = cell(2,2);
container{1,1} = FallSem(1);
container{2,1} = FallSem(2);
container{1,2} = 2;
container{2,2} = 'a';
container
end

solvexf('det');

```

Output

Linear Regression

Commands

```
type forecasting_data.txt;
data = load("-ascii", "forecasting_data.txt");
n = size(data)(1)
AVG = mean(data)
SUM_SQUARE = var(data(:,1))*(n-1)
COV_N = sum((data(:,1) - AVG(1)) .* (data(:,2) - AVG(2)))
b = COV_N / SUM_SQUARE
a = AVG(2) - b * AVG(1)
epsilon = std(data(:,2))
% epsilon = sqrt(sum((data(:,2) - AVG(2)).^2)/(n-1)). This gives a different number from shown.
disp('68% confidence. r not calcualted')
Y_low = a + b*2020 - epsilon
Y_high = a + b*2020 + epsilon
hold on
scatter(data(:,1),data(:,2), 'o')
plot([2012 2020], [2012*b + a, 2020*b + a])
plot([2020 2020], [Y_low, Y_high], 'go-')
plot([2020], [a + b*2020], 'bo')
hold off
xlabel('years')
xlabel('sales')
axis([2012,2021,10,70])
pause
```

Output

```
2013    23
2014    18
2015    35
2016    39
2017    35
2018    37
2019    49
n = 7
AVG =
2016.000      33.714

SUM_SQUARE = 28
COV_N = 116
b = 4.1429
a = -8318.3
epsilon = 10.307
68% confidence. r not calcualted
Y_low = 39.979
Y_high = 60.593
```

